CLAIMS

What is claimed is:

1. A method of automatically controlling execution of an application program having a graphical user interface comprising:

capturing user input data and images displayed by the graphical user interface during a recording phase of execution of the application program;

analyzing the captured user input data and displayed images to generate an execution scenario during the recording phase;

generating simulated user input data based on the execution scenario during a playback phase of execution of the application program and inputting the simulated user input data to the application program;

performing image analysis on images displayed by the graphical user interface as a result of processing the simulated user input data during the playback phase and captured displayed images from the recording phase; and

automatically controlling execution of the application program based at least in part on the image analysis.

2. The method of claim 1, wherein performing image analysis comprises comparing the displayed images captured during the recording phase with displayed images from the playback phase.

3. The method of claim 1, wherein the user input data comprises at least one of keyboard input data and mouse input data.

4. The method of claim 1, wherein analyzing the captured user input data and displayed images in the recording phase comprises identifying actions of the application program corresponding to the captured user input data.

5. The method of claim 1, wherein the execution scenario comprises a script including extended markup language (XML) tags.

6. The method of claim 1, wherein analyzing the captured user input data and displayed images during a recording phase comprises:

determining contours of objects shown in a displayed image;

detecting an object of activity from among the objects; and

detecting additional objects located adjacent to the object of activity.

7. The method of claim 1, wherein performing image analysis comprises

16

determining contours of objects in an image displayed by the graphical user interface during the playback phase and filtering the objects according to size to produce a set of hypotheses for active objects.

8. The method of claim 7, wherein performing image analysis further comprises filtering objects according to space layout to produce a first subset of hypotheses for active objects.

9. The method of claim 7, wherein performing image analysis further comprises filtering objects by content to produce a second subset of hypotheses for active objects.

10. The method of claim 7, wherein performing image analysis further comprises structural filtering of objects to produce a best hypothesis for an active object.

11. The method of claim 7, wherein performing image analysis further comprises recalculating old actions for a new object identified as the active object.

12. An article comprising: a machine readable medium containing instructions, which when executed, result in automatically controlling execution of an application program having a graphical user interface by

capturing user input data and images displayed by the graphical user interface during a recording phase of execution of the application program;

analyzing the captured user input data and displayed images to generate an execution scenario during the recording phase;

generating simulated user input data based on the execution scenario during a playback phase of execution of the application program and inputting the simulated user input data to the application program;

performing image analysis on images displayed by the graphical user interface as a result of processing the simulated user input data during the playback phase and captured displayed images from the recording phase; and

automatically controlling execution of the application program based at least in part on the image analysis.

13. The article of claim 12, wherein instructions for performing image analysis comprise instructions for comparing the displayed images captured during the recording phase with displayed images from the playback phase.

14. The article of claim 12, wherein the user input data comprises at least one of

keyboard input data and mouse input data.

15. The article of claim 12, wherein instructions for analyzing the captured user input data and displayed images in the recording phase comprise instructions for identifying actions of the application program corresponding to the captured user input data.

16. The article of claim 12, wherein the execution scenario comprises a script including extended markup language (XML) tags.

17. The article of claim 12, wherein instructions for analyzing the captured user input data and displayed images during a recording phase comprise instructions for:

determining contours of objects shown in a displayed image;

detecting an object of activity from among the objects; and

detecting additional objects located adjacent to the object of activity.

18. The article of claim 12, wherein instructions for performing image analysis comprise instructions for determining contours of objects in an image displayed by the graphical user interface during the playback phase and filtering the objects according to size to produce a set of hypotheses for active objects.

19. The article of claim 18, wherein instructions for performing image analysis further comprise instructions for filtering objects according to space layout to produce a first subset of hypotheses for active objects.

20. The method of claim 18, wherein instructions for performing image analysis further comprise instructions for filtering objects by content to produce a second subset of hypotheses for active objects.

21. The article of claim 18, wherein instructions for performing image analysis further comprise instructions for structural filtering of objects to produce a best hypothesis for an active object.

22. The article of claim 18, wherein instructions for performing image analysis further comprise instructions for recalculating old actions for a new object identified as the active object.

23. A cognitive control framework system for automatically controlling execution of an application program having a graphical user interface comprising:

a recording component adapted to capture user input data and images displayed by the graphical user interface during a recording phase of execution of the application

18

program, and to analyze the captured user input data and displayed images to generate an execution scenario during the recording phase,

a playback component adapted to generate simulated user input data based on the execution scenario during a playback phase of execution of the application program, to input the simulated user input data to the application program, to perform image analysis on images displayed by the graphical user interface as a result of processing the simulated user input data during the playback phase and captured displayed images from the recording phase; and to automatically control execution of the application program based at least in part on the image analysis.

24. The system of claim 23, wherein the playback component is adapted to compare the displayed images captured during the recording phase with displayed images from the playback phase.

25. The system of claim 23, wherein the recording component is adapted to identify actions of the application program corresponding to the captured user input data.

26. The system of claim 23, wherein the recording component is adapted to determine contours of objects shown in a displayed image, detect an object of activity from among the objects, and detect additional objects located adjacent to the object of activity.

27. The system of claim 23, wherein the playback component is adapted to determine contours of objects in an image displayed by the graphical user interface during the playback phase and filter the objects according to size to produce a set of hypotheses for active objects.

28. The system of claim 27, wherein the playback component is adapted to filter objects according to space layout to produce a first subset of hypotheses for active objects.

29. The system of claim 27, wherein the playback component is adapted to filter objects by content to produce a second subset of hypotheses for active objects.

30. The system of claim 27, wherein the playback component is adapted to structurally filter objects to produce a best hypothesis for an active object.